REMARKS

Claims 1 - 39 are pending in this application. Claims 1, 9, 19 and 31 are independent.

The Office has indicated claims 7 and 17 would be allowable if the subject matter of the base claims and the claims from which they depend were incorporated into the respective claims

Claims 1-6, 8-16, 18, 20-30, and 32-39 stand rejected under 35 U.S.C. § 103(a).

Reconsideration in view of the above-listed amendments and the following remarks is respectfully requested.

Allowable Subject Matter

Applicant acknowledges the Office's determination that claims 7 and 17 recite patentable subject matter. Applicant proposes amending independent claims 1, 9, and 19 to incorporate concepts from claims 7 and 17.

Rejections under 35 U.S.C. § 103

Claims 1 – 4, 6, 8 – 14, 16, 18-24, 26 – 30, 31-37, and 39 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over U.S. patent 5,261,085 (hereinafter "085 patent") in view of Paxos Made Simple (hereinafter "Paxos") from Applicant's IDS dated December 30, 2003 and U.S. patent 6,532,494 (hereinafter "494 patent").

Claims 5, 15 and 25 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the 085 patent in view of Paxos and the 494 patnet in further view of U.S. patent publication 2003/0227392 (the "392 application").

Claim 38 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over the 085 patent in view of Paxos and the 494 patent and in view of U.S. patent publication 2002/0112198 (the "198 application").

Reconsideration is respectfully requested.

Applicant notes in the patent application specification that:

> [I]n the event of conflicts, the Fast Paxos algorithm can, by performing the first phase of the standard Paxos algorithm, introduce more message delays than would have otherwise been present if the system 10 had been using the standard

> Paxos algorithm all along. **Because conflicts can arise frequently in an environment in which more than once device may seek to act as a client, a reduced message delay consensus algorithm such as Fast Paxos may not provide the expected efficiencies unless it can continue operating properly in the face of conflicting client proposals.** (¶ [0108]).

Applicant has therefore disclosed:

> [A] system can implement a reduced message delay consensus **algorithm that is conflict tolerant**. Turning to FIG. 8a, an exemplary environment is shown comprising one client device 20, and additional devices 11-15 that are both the constituent devices of the distributed computing system 10, and can act as clients of the system 10. Furthermore, as shown, **each of the devices 11-15 and the client 20 can be assigned a client identifier**. In one embodiment contemplated by the present invention, the constituent devices of the system 10 essentially **vote for a combination of a proposed function, and the particular device that proposed the function**. Thus, **while a device might vote for a proposed function from one device, it might not vote for the same proposed function if it was proposed by a different device**. As will be shown in more detail below, **a reference to the identifier of the device proposing the function can help provide conflict tolerance**. (¶ [0110]).
>
> As will be known to those skilled in the art, the selection and assignment of client identifiers to the clients of the system 10 can occur through any number of mechanisms, and the embodiments of the present invention are not dependent upon, nor are they intended to be limited to, any particular mechanism. By way of example only, the class identifiers could be assigned through a registration process, such as with a central registration server. Alternatively, the class identifiers could be assigned based on unique properties of the devices, such as the exact time at which they joined the distributed computing system, their MAC address, or the like. Yet another alternative would be hard code identifiers into the software implementing the above described algorithms, or into particular hardware elements, such as the ROM 131, network interface 170, or the like. (¶ [0111]).

Applicant still further explains:

> [0112] Furthermore, as will be apparent to those skilled in the art from the following descriptions, the ordering of the client identifiers can be arbitrary. **Thus, client identifiers can be ordered in the manner described below, with a numerically larger value client identifier being more <u>dominant</u> than a numerically lower value client identifier. Alternatively, a numerically larger value client identifier can be less <u>dominant</u> than a numerically lower value client identifier. Similarly, client identifiers of a particular type, such as beginning or ending with a particular value, can be more dominant than client identifiers that do not begin or end with the particular value. In whichever manner the client identifiers are ordered, the client identifier assigned to the client device 20, which does not also act as a device implementing the distributed system 10, can be the least <u>dominant</u> client identifier, such that the client identifiers assigned to devices 11-15 are all more dominant than the client identifier assigned to the client 20**. (¶ [0112]).

Claim 1 recites:

> A method for selecting a value in a distributed computing system **using a fault tolerant consensus algorithm**, the method comprising:
>
> receiving at a computing device from a first client a first message comprising a first proposed value and a first client identifier corresponding to the first client;
>
> provisionally voting at the computing device for the first proposed value;
>
> transmitting from the computing device a first indication of the provisional voting for the first proposed value to one or more devices;
>
> transmitting from the computing device a first result of the provisional voting for the first proposed value to the first client,
>
> **wherein the voting for the first proposed value, the transmitting the first indication of the voting for the first proposed value, and the transmitting the first result are <u>not</u> performed <u>if</u> a second message had previously been received at the computing device from a second client, the second message comprising a second proposed value and a second client identifier corresponding to a second client, <u>the second client identifier being more dominant than the first</u>**

> **<u>client identifier, and the second proposed value having been previously provisionally voted for</u>**; and
>
> **receiving at the computing device from a third client a third message comprising a third proposed value and a third client identifier corresponding to the third client, the computing device provisionally voting for the third proposed value and transmitting a third indication of the provisional voting for the third proposed value <u>if the third client identifier is dominant as compared to the first client identifier and the third client identifier</u>**.

In order for a reference to anticipate this claim, or a set of references to render the claim obvious, the recited language and its combination in the recited method must be taught by the prior art. The undersigned respectfully submits that the cited references do not teach the emphasized language and cannot possibly teach or suggest the recited method.

The Office appears to acknowledge that the 085 patent does not disclose:

> **the voting for the first proposed value, the transmitting the first indication of the voting for the first proposed value, and the transmitting the first result are <u>not</u> performed <u>if</u> a second message had previously been received at the computing device from a second client, the second message comprising a second proposed value and a second client identifier corresponding to a second client, <u>the second client identifier being more dominant than the first client identifier, and the second proposed value having been previously provisionally voted for</u>**.

Indeed, the 085 patent discloses messages being received from a single "client," namely the "leader." Furthermore, the 085 patent does not disclose:

> **receiving at the computing device from a third client a third message comprising a third proposed value and a third client identifier corresponding to the third client, the computing device provisionally voting for the third proposed value and transmitting a third indication of the provisional voting for the third proposed value <u>if the third client identifier is dominant as compared to the first client identifier and the third client identifier</u>**

The 085 patent simply does not disclose "[a] method for selecting a value in a distributed computing system **using a fault tolerant consensus algorithm**."

Paxos likewise does not disclose the emphasized claim language. Applicant acknowledges the existence of Paxos, and, in fact, discusses the Paxos algorithm and another algorithm known as the Fast Paxos algorithm in the present application. (See e.g., ¶¶ [0008] – [0013]). Applicant has noted that both algorithms have limitations. In particular, the Paxos algorithm introduces delays:

> However, the Paxos algorithm adds message delays between when a client sends a request for the distributed system to execute a command, and when the client receives the results from the execution that command. Specifically, even if the client transmits a request to a leader, and even if the leader has already learned of previously voted on proposals, and thus has completed the first phase of the Paxos algorithm, there can still be two or more message delays between the transmission of the request from the client, and the transmission of the results to the client. **Furthermore, the Paxos algorithm <u>can require the presence of a leader device that receives client requests and determines the appropriate functions to submit for a vote to the devices of the distributed computing system</u>. Should such a leader device fail, a new leader may not take its place immediately, leaving the distributed computing system idle and the client waiting for a response to its requests.** (Application, ¶ [0011]).

Applicant has noted that the Fast Paxos algorithm **cannot tolerate a conflict among two or more clients**.

> [T]he Fast Paxos algorithm cannot tolerate a conflict among two or more clients. Specifically, if two or more clients propose different functions at approximately the same time, the devices may be unable to choose between the different functions. In such a case, the system must stop using the Fast Paxos algorithm and return to the regular Paxos algorithm, with the leader beginning with the first phase, in an effort to resolve the discrepancy among the devices in the system. In such a case, the two or more clients that submitted the conflicting proposals may experience an even greater delay in receiving their responses than if the system had never attempted to operate using the Fast Paxos algorithm. (Application, ¶ [0013]).

Accordingly, Paxos does not disclose "[a] method for selecting a value in a distributed computing system **using a fault tolerant consensus algorithm**" as recited in the claims. Paxos certainly does not disclose:

> **the voting for the first proposed value, the transmitting the first indication of the voting for the first proposed value, and the transmitting the first result are <u>not</u> performed <u>if</u> a second message had previously been received at the computing device from a second client, the second message comprising a second proposed value and a second client identifier corresponding to a second client, <u>the second client identifier being more dominant than the first client identifier, and the second proposed value having been previously provisionally voted for</u>**, and
>
> **receiving at the computing device from a third client a third message comprising a third proposed value and a third client identifier corresponding to the third client, the computing device provisionally voting for the third proposed value and transmitting a third indication of the provisional voting for the third proposed value <u>if the third client identifier is dominant as compared to the first client identifier and the third client identifier.</u>**

Rather, in the system disclosed by Paxos, proposals are received that are identified by a *proposal number*. The proposal numbers of Paxos do not comprise "a first proposed valued and **a first client identifier corresponding to the first client,**" "a second proposed value and a **second client identifier corresponding to a second client,**" and "a third proposed value and **a third client identifier corresponding to the third client.**"

Furthermore, the Paxos algorithm does not disclose "not" performing "the voting for the first proposed value, the transmitting the first indication of the voting for the first proposed value, and the transmitting the first result" **if** "a second message had previously been received . . . **the second message comprising a second proposed value and a second client identifier corresponding to a second client, <u>the second client identifier being more dominant than the first client identifier</u> and the second proposed value having been previously voted for**." Likewise, Paxos similarly does not disclose or suggest "**provisionally voting for the third proposed value and transmitting a third indication of**

the provisional voting for the third proposed value <u>if the third client identifier is dominant as compared to the first client identifier and the third client identifier</u>." Rather, the Paxos algorithm relies on proposal numbers to determine what proposal to execute. Paxos does not determine or consider "the second client identifier being more dominant than the first client identifier." Indeed, the concept is entirely absent.

The Office notes that Paxos discloses selecting a value in a network where there are multiple acceptors. (Office Action, p. 4). In particular, Paxos discloses that a value is chosen when a large enough number of acceptors accept the value.

> A proposer sends a proposed value to a set of acceptors. An acceptor may accept the proposed value. The value is chosen when a large enough set of acceptors have accepted it.

But selecting a value because enough acceptors have accepted the value is not the same or even similar to "not" performing various activities "if" "a second message has been previously received . . . **the second message comprising a second proposed value and a second client identifier corresponding to a second client, <u>the second client identifier being more dominant than the first client identifier</u>**." Rather, in Paxos, the selected value is based on whether a plurality of other acceptors have also accepted. The selection is not determined by whether a client identifier from a second client is more dominant than a first.

The Office further notes that Paxos discloses an acceptor accepting a proposed value if it has not already responded to a request having a greater number. (Office Action, p. 4).

> An acceptor can accept a proposal numbered n if it has not responded to a prepare request having a number greater than n.

Thus, Paxos teaches selecting a proposal based upon a ***proposal number***. The referenced section of Paxos does not indicate that a proposal is selected based upon a ***client identifier*** as recited in the claim. There is not indication that a client identifier is even available to be considered in the Paxos algorithm.

The 494 patent discloses methods for monitoring network clusters. In the disclosed methods, a cluster manager uses disk based messaging to manage the operation of the cluster. Each node within the cluster must have access to a shared disk to operate within the cluster. If a node fails to receive a heartbeat message from its predecessor in the loop, it initiates a

cluster reconfiguration by sending a reconfiguration message to the other nodes in the cluster. The quorumless cluster can also include a common storage for a cluster definition. Each node may provide a proposed change to the cluster definition, however, **only a single coordinator node may update the cluster definition and apply the suggested changes**. (494 patent at Abstract).

Thus, the 494 patent discloses methods that provide for **only a single coordinator to make changes to a definition**. The 494 patent does not disclose "[a] method for selecting a value in a distributed computing system **using a fault tolerant consensus algorithm**" as recited in the claims. The 494 patent certainly does not disclose:

> **the voting for the first proposed value, the transmitting the first indication of the voting for the first proposed value, and the transmitting the first result are <u>not</u> performed <u>if</u> a second message had previously been received at the computing device from a second client, the second message comprising a second proposed value and a second client identifier corresponding to a second client, <u>the second client identifier being more dominant than the first client identifier, and the second proposed value having been previously provisionally voted for</u>**, and
>
> **receiving at the computing device from a third client a third message comprising a third proposed value and a third client identifier corresponding to the third client, the computing device provisionally voting for the third proposed value and transmitting a third indication of the provisional voting for the third proposed value <u>if the third client identifier is dominant as compared to the first client identifier and the third client identifier</u>**

Therefore, because neither the 085 patent nor Paxos disclose the emphasized claim language, even in combination the references cannot be said to disclose or suggest the recited combination of claim 1.

Claims 9, 19, and 31 recite language that is different from claim 1. However, for reasons analogous to those discussed above in connection with claim 1, these claims are likewise patentable over the cited references. All dependent claims are likewise patentable for being dependent upon a patentable base claim.

Reconsideration and withdrawal of the rejections under 35 U.S.C. § 103 is respectfully solicited.

CONCLUSION

The undersigned respectfully submits that pending claims are allowable and the application is in condition for allowance. A Notice of Allowance is respectfully solicited.

Examiner Sciacca is invited to call the undersigned in the event a telephone interview will advance prosecution of this application.

Date: February 25, 2010

/John E. McGlynn/
John E. McGlynn
Registration No. 42,863

Woodcock Washburn LLP
Cira Centre
2929 Arch Street, 12th Floor
Philadelphia, PA 19104-2891
Telephone: (215) 568-3100
Facsimile: (215) 568-3439